EPSON®

SPC8106F0C LCD/CRT VGA Controller

BIOS Functional Specification

Document Number: X12-SP-002-05

Information in this document is subject to change without notice. You may download and use this document, but only for your own use in evaluating Seiko Epson/EPSON products. You may not modify the document. Epson Research and Development, Inc. disclaims any representation that the contents of this document are accurate or current. The Programs/Technologies described in this document may contain material protected under U.S. and/or International Patent laws.

THIS PAGE LEFT BLANK

Table of Contents

THIS PAGE LEFT BLANK

1 INTRODUCTION

1.1 Scope and Objectives

This is the Functional Specification for the SPC8104F0A BIOS. This document specifies the functions, structures and characteristics of the SPC8106F0C BIOS. It is intended for use by persons familiar with VGA BIOS functions and describes differences between the SPC8106F0C and a standard VGA BIOS. In addition this manual documents some of the behavior characteristics and structure of the BIOS.

This manual consists of several sections which include:

- Features
- Video Modes
- Main BIOS Summary
- VESA Function Summary
- SOLLEX Function Summary
- Physical layout of the BIOS
- and a series of Appendices with chip specific information

2 FEATURES

2.1 Technology

- 32 KB EPROM maximum size
- Microsoft MASM 5.1 compatible source

2.2 System and System Compatibility

- ISA machine architecture
- support for VESA VBE Core Functions version 1.2 and SOLLEX BIOS extensions
- support for VESA VBE/PM power management functions version 1.0 for BIOS version 1.14 or later
- five software power-save modes
- 3C3h or 46E8h video enable register supported (configurable)
- support for all standard IBM defined VGA modes, including modes 07h and 0Fh
- support for VESA VBE/DDC display data channel functions version 1.0 for BIOS version 1.2 or later
- selectable gray-scaling
- normal/reverse and autoswitch text/graphics of display polarity support
- autocenter support
- IRQ enable/disable support
- Selectable Cursor Blink rate
- 400 to 475 scan line expansion in text and graphics modes

2.3 Display support

- supports a variety of single/dual LCD panels of various resolutions from 320 x 200 to 640 x 480, based on MD line inputs
- supports CRT with External RAMDAC attached

2.4 Video Modes

2.4.1 LCD Display Modes

Mode No.	Mode Type	Font	Characters	Resolution	Displayed Pixels	Gray Shades	Colors	Memory Segment
0	Text	8 x 8	40 x 25	320 x 200	640 x 400	16	16	B800
0+	Text	8 x 14	40 x 25	320 x 350	640 x 350	16	16	B800
0++	Text	8 x 16	40 x 25	320 x 400	640 x 400	16	16	B800
1	Text	8 x 8	40 x 25	320 x 200	640 x 400	16	16	B800
1+	Text	8 x 14	40 x 25	320 x 350	640 x 350	16	16	B800
1++	Text	8 x 16	40 x 25	320 x 400	640 x 400	16	16	B800
2	Text	8 x 8	80 x 25	640 x 200	640 x 400	16	16	B800
2+	Text	8 x 14	80 x 25	640 x 350	640 x 350	16	16	B800
2++	Text	8 x 16	80 x 25	640 x 400	640 x 400	16	16	B800
3	Text	8 x 8	80 x 25	640 x 200	640 x 400	16	16	B800
3+	Text	8 x 14	80 x 25	640 x 350	640 x 350	16	16	B800
3++	Text	8 x 16	80 x 25	640 x 400	640 x 400	16	16	B800
4	Graphics	N/A	N/A	320 x 200	640 x 400	4	4	B800
5	Graphics	N/A	N/A	320 x 200	640 x 400	4	4	B800
6	Graphics	N/A	N/A	640 x 200	640 x 400	2	2	B800
7	Text	8 x 14	80 x 25	640 x 350	640 x 350	2	2	B000
7+	Text	8 x 16	80 x 25	640 x 400	640 x 400	2	2	B000
0D	Graphics	N/A	N/A	320 x 200	640 x 400	16	16	A000
0E	Graphics	N/A	N/A	640 x 200	640 x 400	16	16	A000
0F	Graphics	N/A	N/A	640 x 350	640 x 350	2	2	A000
10	Graphics	N/A	N/A	640 x 350	640 x 350	16	16	A000
11	Graphics	N/A	N/A	640 x 480	640 x 480	2	2	A000
12	Graphics	N/A	N/A	640 x 480	640 x 480	16	16	A000
13	Graphics	N/A	N/A	320 x 200	640 x 400	64	256	A000
100	Graphics	N/A	N/A	640 x 400	640 x 400	64	256	A000
101	Graphics	N/A	N/A	640 x 480	640 x 480	64	256	A000
108	Text	8 x 8	80 x 60	640 x 480	640 x 480	16	16	B800

2.4.2 CRT Display Modes

Mode No.	Mode Type	Font	Characters	Resolution	Displayed Pixels	Colors	Memory Segment
0	Text	8 x 8	40 x 25	320 x 200	640 x 400	16	B800
0+	Text	8 x 14	40 x 25	320 x 350	640 x 350	16	B800
0++	Text	9 x 16	40 x 25	360 x 400	720 x 400	16	B800
1	Text	8 x 8	40 x 25	320 x 200	640 x 400	16	B800
1+	Text	8 x 14	40 x 25	320 x 350	640 x 350	16	B800
1++	Text	9 x 16	40 x 25	360 x 400	720 x 400	16	B800
2	Text	8 x 8	80 x 25	640 x 200	640 x 400	16	B800
2+	Text	8 x 14	80 x 25	640 x 350	640 x 350	16	B800
2++	Text	9 x 16	80 x 25	720 x 400	640 x 400	16	B800
3	Text	8 x 8	80 x 25	640 x 200	640 x 400	16	B800
3+	Text	8 x 14	80 x 25	640 x 350	640 x 350	16	B800
3++	Text	9 x 16	80 x 25	720 x 400	640 x 400	16	B800
4	Graphics	N/A	N/A	320 x 200	640 x 400	4	B800
5	Graphics	N/A	N/A	320 x 200	640 x 400	4	B800
6	Graphics	N/A	N/A	640 x 200	640 x 400	2	B800
7	Text	8 x 14	80 x 25	640 x 350	640 x 350	2	B000
7+	Text	9 x 16	80 x 25	720 x 400	720 x 400	2	B000
0D	Graphics	N/A	N/A	320 x 200	640 x 400	16	A000
0E	Graphics	N/A	N/A	640 x 200	640 x 400	16	A000
0F	Graphics	N/A	N/A	640 x 350	640 x 350	2	A000
10	Graphics	N/A	N/A	640 x 350	640 x 350	16	A000
11	Graphics	N/A	N/A	640 x 480	640 x 480	2	A000
12	Graphics	N/A	N/A	640 x 480	640 x 480	16	A000
13	Graphics	N/A	N/A	320 x 200	640 x 400	256	A000
100	Graphics	N/A	N/A	640 x 400	640 x 400	256	A000
101	Graphics	N/A	N/A	640 x 480	640 x 480	256	A000
108	Text	8 x 8	80 x 60	640 x 480	640 x 480	16	B800

2.5 Simultaneous Display Support

Simultaneous Display is a feature in the SPC8106F0C that allows simultaneous display of both the CRT and LCD panel. Refer to the following table for supported video modes and limitations.

Note
Simultaneous Display is not supported when using a Dual panel / Dual drive LCD.

Note
All non-480 line modes (when using a 480 line panel) will show a wrap-around effect if Simultaneous Display is enabled. For example: display a 400 line mode on a 480 line panel and enable Simultaneous Display. You would see the top 78 lines duplicated on the bottom 78 lines (wrap-around affect).

Note
If using a panel with less than 480 vertical lines, Simultaneous Display will be supported, however, the maximum LCD frame-rate may be violated. Therefore, the specific panel should be referenced.

Note
If supporting a TFT panel requiring CRT-like timing (AUX[00] bit 5=1 and AUX[0B] bit 1=1) the panel handles the 350, 400 and 480 line modes, providing screen positioning internally. As the result of this direct support, Simultaneous Display is supported for all standard VGA and some extended modes.

If supporting a TFT panel requiring LCD-like timing (AUX[00] bit 5=1 and AUX[0B] bit 1=0) the Simultaneous Display mode is not supported.

Mode No.	Single Panel (640x480)	TFT - Color 9/12-bit (640x480) AUX[00] bit 5=1 AUX[0B] bit 1=1
0	No	Yes
0+	No	Yes
0++	No	Yes
1	No	Yes
1+	No	Yes
1++	No	Yes
2	No	Yes
2+	No	Yes
2++	No	Yes
3	No	Yes
3+	No	Yes
3++	No	Yes
4	No	Yes
5	No	Yes
6	No	Yes
7	No	Yes
7+	No	Yes
0D	No	Yes
0E	No	Yes
0F	No	Yes
10	No	Yes
11	Yes	Yes
12	Yes	Yes
13	No	Yes
100	No	Yes
101	Yes	Yes
108	Yes	Yes

3 OVERVIEW DESCRIPTION

The SPC8106F0C is a single chip 5 Volt LCD/CRT video controller based on VGA architecture and optimized for driving an LCD panel display. VGA standard mode functionality is supported using standard IBM VGA parameters. A proprietary 256 x 6-bit gray-scale lookup table is provided to allow remapping of the 64 possible gray shades displayed on a monochrome LCD panel. A 256 x 12-bit lookup table (4 bits per primary) provides 256 out of a possible 4096 colors on a color LCD panel.

The target market for this device is notebook computers, or other specialized consumer products. The ability to run all VGA software on a 640x480 LCD panel display is the major design consideration, therefore the BIOS must perform the same functions that the IBM VGA BIOS performs within the limitations of the SPC8106F0C.

The BIOS is divided into 3 major sections: the main BIOS, the VESA Extensions and the Sollex Extensions. It requires 32 KB of EPROM space, decoded by the system board.

3.1 Main BIOS

The main BIOS contains the core VGA compatible information. It requires 24 KB of the total 32 KB of EPROM space. This main BIOS is responsible for the initialization of the chip and for performing the IBM compatible function calls. It contains the Video Parameter Tables and the Character Tables.

3.2 VESA Extensions

The VESA Extensions are found in the last 8 KB of the 32 KB. This contains the VESA compatible functions as defined by the Video Electronics Standards Association. At time of printing the BIOS conforms to the Video BIOS Extensions Standard 1.2. These VESA functions are responsible for setting non-IBM modes and supplying functions for mode information. More information on VESA can be obtained by contacting the Video Electronics Standards Association located in San Jose, CA.

3.3 Sollex Extensions

The Sollex Extensions are defined by Seiko Epson Corporation to augment the functionality of the BIOS to include panel and power down functions. This also resides in the last 8 KB. More information on Sollex can be found in the *SOLLEX Specification* Drawing Office No. S03-SP-001-xx.

4 MAIN BIOS FUNCTION SUMMARY

4.1 Supported BIOS Functions

These functions are the IBM defined functions that are supported on all VGA compatible products. These functions are supported in the SPC8106F0C BIOS with the noted exceptions. These functions are called using the standard INT 10h interface. To call these functions:

```
MOV AH, function number
MOV other register, other parameters
INT 10H
```

Function 00h - Set Video Mode

```
Input:     AH=00h   Set Video Mode
           AL       Video mode (bit 7 set prevents VRAM c lear)
Return:    n/a
```

Function 01h - Set Cursor Type

```
Input:     AH=01h   Set Cursor Type
           CH       Cursor start scan
           CL       Cursor end scan
Return:    n/a
```

Function 02h - Set Cursor Position

```
Input:     AH=02h   Set Cursor Position
           BH       Page number
           DL       Column (0-x)
           DH       Row (0-x)
Return:    n/a
Destroyed: AX,SI
```

Function 03h - Read Cursor Position

```
Input:     AH=03h   Read Cursor Position
           BH       Page number
Return:    CX       Current cursor mode
           DX       Current cursor position
```

Function 04h - Read Lightpen Position (Unsupported On VGA)

```
Input:     AH=04h   Read Lightpen Position
Return:    AH       Lightpen status (0=none, 1=active)
                    If AH=1 then:
                             BX - Pixel column
                             CX - Scan line
                             DX - Character row/column
```

Function 05h - Select Active Display Page

Input:	AH=05h	Select Active Display Page
	AL	New page number
Return:	n/a	

Function 06h - Scroll Active Page Up

Input:	AH=06h	Scroll Active Page Up
	AL	Lines to scroll (0=blank window)
	BH	New line(s) attribute
	CX	Top-left corner of scroll window
	DX	Bottom-right corner of scroll window
Return:	n/a	
Destroyed:	AX, SI, DI, (and DS if text modes)	

Function 07h - Scroll Active Page Down

Input:	AH=07h	Scroll Active Page Down
	AL	Lines to scroll (0=blank window)
	BH	New line(s) attribute
	CX	Top-left corner of scroll window
	DX	Bottom-right corner of scroll window
Return:	n/a	

Function 08h - Read Character/attribute at Cursor Position

Input:	AH=08h	Read Character/attribute at Cursor Posit ion
	BH	Page number
Return:	AL	Character read
	AH	Attribute read
Destroyed:	AX, SI, DI, (and DS if text modes)	

Function 09h - Write Character/attribute at Cursor Position

Input:	AH=09h	Write Character/attribute at Cursor Posi tion
	AL	Character to write
	BL	Character attribute/color (b7 set for XOR)
	BH	Page number (Background color in Mode 13)
	CX	Character count
Return:	n/a	
Destroyed:	AX, SI, DI, (and DS if text modes)	

Function 0Ah - Write Character Only at Cursor Position

Input:	AH=0Ah	Write Character only at Cursor Position
	AL	Character to write
	BH	Page number
	CX	Character count
Return:	n/a	
Destroyed:	AX, SI, DI, (and DS if text modes)	

Function 0Bh - Set Color Palette

Input:	AH=0Bh	Set Color Palette	
	BH=0	(selects background color)	
		BL=0-Fh	Background color
	BH=1	(selects palette)	
		BL=0	(Green, Red, Brown)
		BL=1	(Cyan, Magenta, White)
Return:	n/a		

Function 0Ch - Write Dot

Input:	AH=0Ch	Write Dot
	AL	Color (b7 set for XOR)
	BH	Page number
	CX	Column
	DX	Row
Return:	n/a	

Function 0Dh - Read Dot

Input:	AH=0Dh	Read Dot
	BH	Page number
	CX	Column
	DX	Row
Return:	AL	Dot color

Function 0Eh - Write TTY Character to Active Page

Input:	AH=0Eh	Write TTY Character to Active Page
	AL	Character (CR, LF, BS, and BELL accepted)
	BL	Color in graphics mode
Return:	n/a	

Function 0Fh - Get Current Video State

Input:	AH=0Fh	Get Current Video State
Return:	AL	Current video mode
	AH	Number of columns
	BH	Current page number

Function 10h - Palette Functions

Input:	AH=10h	Palette Functions
	AL=00	Set palette register
	BL	Palette register
	BH	Value to be set
Return:	n/a	

Input:	AH=10h	Palette Functions
	AL=01	Set overscan register
	BH	Value to be set
Return:	n/a	

Input:	AH=10h	Palette Functions
	AL=02	Set all palette registers and overscan
	ES:DX	Pointer to 17-byte table
Return:	n/a	

Input:	AH=10h	Palette Functions
	AL=03	Toggle intensity/blink bit
	BL	1=Blink, 0=Intensity
Return:	n/a	

Input:	AH=10h	Palette Functions
	AL=07	Get palette register
	BL	Palette register
Return:	BH	Palette register value

Input:	AH=10h	Palette Functions
	AL=08	Get overscan register
Return:	BH	Overscan register value

Input:	AH=10h	Palette Functions
	AL=09	Get all palette registers and overscan
	ES:DX	Pointer to 17-byte table
Return:	n/a	

Input:	AH=10h	Palette Functions
	AL=10h	Set DAC color register
	BX	Color register
	DH:CH:CL	Red, Green, Blue data
Return:	n/a	

```
Input:     AH=10h     Palette Functions
           AL=12h     Set block of DAC registers
           BX         Start color register
           CX         Number of registers
           ES:DX      Pointer to RGB table
Return:    n/a

Input:     AH=10h     Palette Functions
           AL=13h     Select color page
           BL         Paging function (0-1)
                      00       Select paging mode
                      01       Select page
           BH         If BL=0 (0=4 of 64, 1=16 of 16)
                      If BL=1 (Page number 0-3, 0-15)
Return:    n/a

Input:     AH=10h     Palette Functions
           AL=15h     Get DAC color register
           BX         Color register
Return:    DH         Red value
           CH         Green value
           CL         Blue value

Input:     AH=10h     Palette Functions
           AL=17h     Get block of DAC registers
           BX         Start color register
           CX         Number of registers
           ES:DX      Pointer to RGB table
Return:    n/a

Input:     AH=10h     Palette Functions
           AL=18h     Set PEL Mask (VGA Undocumented)
           BL         PEL Mask to write
Return:    n/a

Input:     AH=10h     Palette Functions
           AL=19h     Get PEL Mask (VGA Undocumented!)
           BX         Returned PEL Mask value
Return:    n/a

Input:     AH=10h     Palette Functions
           AL=1Ah     Get current color page
           BL         Returned paging mode
           BH         Returned page number
Return:    n/a
```

```
Input:     AH=10h   Palette Functions
           AL=1Bh   Convert all DAC registers to gray-scale
           BX       Start color register
           CX       Number of registers
Return:    n/a
```

Note
Overscan Subfunction 01 incorrectly puts the data into the parameter save area offset + 11h to be consistent with IBM code.

Function 11h - Character Generator Control

```
           AL                Character generator function (0-30h)
                             0x - Alpha load (x=0-4)
                             1x - Alpha load, recalculated (x=0-2, 4)
                             2x - Graphics load (x=0-4)
                             30 - Return information

Input:     AH=11h   Character generator function
           AL=00 or AL=10h (user alpha load):
           BL       Block to load
           BH       Points
           CX       Character count
           DX       Character offset
           ES:BP    Font table pointer
Return:    n/a

Input:     AH=11h   Character generator function
           AL=01 or AL=11h (ROM 8x14 set):
           BL       Block to load
Return:    n/a

Input:     AH=11h   Character generator function
           AL=02 or AL=12h (ROM 8x8 set):
           BL       Block to load
Return:    n/a

Input:     AH=11h   Character generator function
           AL=04 or AL=14h (ROM 8x16 set):
           BL       Block to load
Return:    n/a

Input:     AH=11h   Character generator function
           AL=03    (set active block):
           BL       Value for sequencer register 3
Return:    n/a
```

Input:	AH=11h	Character generator function
	AL=20h	(user graphics characters):
	ES:BP	Font table pointer (chars 128-255)
Return:	n/a	

Input:	AH=11h	Character generator function
	AL=21h	(user graphics load):
	BL	Rows select (0=user, 1=14, 2=25, 3=43)
	CX	Points
	DL	Rows Input: BL=0
	ES:BP	Font table pointer
Return:	n/a	

Input:	AH=11h	Character generator function
	AL=22h	(ROM 8x14 set):
	BL	Rows select
Return:	n/a	

Input:	AH=11h	Character generator function
	AL=23h	(ROM 8x8 set):
	BL	Rows select
Return:	n/a	

Input:	AH=11h	Character generator function
	AL=24h	(ROM 8x16 set):
	BL	Rows select
Return:	n/a	

Input:	AH=11h	Character generator function
	AL=30h	(return information):
	BH	Function request (0-7) 0 - Get INT 1F pointer 1 - Get INT 43 pointer 2 - Get ROM 8x14 pointer 3 - Get ROM 8x8 pointer 4 - Get ROM 8x8 pointer (128-255) 5 - Get 9x14 fudge table pointer 6 - Get ROM 8x16 pointer 7 - Get 9x16 fudge table pointer (fudge font never used, but point size returned should be valid)
Return:	CX	Points
	DL	Rows
	ES:BP	Table pointer

Function 12h - Miscellaneous Functions

Input:	AH=12h	Miscellaneous Function
	BL=10h	Return EGA information
Return:	BL	Memory (0=64K, 1=128K, 2=192K, 3=256K)
	BH	0 = color mode active, 1 = mono mode active
	CL	Switch settings
	CH	Feature bits
Input:	AH=12h	Miscellaneous Function
	BL=20h	Select EGA print screen routine
Return:	n/a	
Input:	AH=12h	Miscellaneous Function
	AL=30h	Set alpha mode scan count
	AL	0=200, 1=350, 2=400 scans
Return:	AL=12h	
Input:	AH=12h	Miscellaneous Function
	BL=31h	Palette load on mode set
	AL	0=enable, 1=disable
Return:	AL=12h	
Input:	AH=12h	Miscellaneous Function
	BL=32h	Video control
	AL	0=enable, 1=disable
Return:	AL=12h	
Input:	AH=12h	Miscellaneous Function
	BL=33h	gray-scale summing
	AL	0=enable, 1=disable
Return:	AL=12h	
Input:	AH=12h	Miscellaneous Function
	BL=34h	Cursor emulation control
	AL	0=enable, 1=disable
Return:	AL=12h	
Input:	AH=12h	Miscellaneous Function
	BL=35h	Display control
	AL	Function request
		00 - Adapter off (initial)
		01 - Planar on (initial)
		02 - Active off
		03 - Inactive on
		ES:DX Pointer to 128-byte buffer

```
Return:    n/a

Input:     AH=12h   Miscellaneous Function
           BL=36h   Video data control
           AL       0=enable, 1=disable
Return:    AL=12h
```

Function 13h - Write String Functions

```
Input:     AH=13h   Write String function
           AL       String function
                    0 - BL=attribute, string=char, char,...
                    1 - BL=attribute, string=char, char, ... cursor  moved
                    2 - String=char, attr, char, attr...
                    3 - String=char, attr, char, attr ... cursor mov ed
           BL       Attribute (if AL=0 or 1)
           BH       Page number
           CX       Character count
           DX       Start cursor position
           ES:BP    String pointer
Return:    n/a
```

Note

If any scrolling occurs, the active page will be scrolled, not the requested page! This "feature" is also present in IBM's code and it has been determined that it is better to reproduce this in our code for compatibility.

Function 14h To Function 19h - Null Functions

Functions 14h to 19h are reserved by VGA definition. Any requests of these functions will just simply return and nothing happens.

```
Input:     n/a
Return:    n/a
```

Function 1Ah - Read/Write Display Combination Code

```
Input:     AH=1Ah   Read/write Display Combination Code
           AL       0=read, 1=write
           BL       Active DCC (r/w)
           BH       Alternate DCC (r/w)
Return:    AL=1Ah
           BX       Set as above
```

Function 1Bh - Return Functionality/State Information

```
Input:     AH=1Bh   Return Functionality/State Information
           BX       Implementation type (must be 00)
           ES:DI    Pointer to buffer
Return:    AL=1Bh
           ES:DI    Contains information
```

The format of the information block is as follows:

```
Offset     Type     Description
00         DD       Pointer to static functionality ta ble
04         DB       CRT video mode-------------------+
05         DW       CRT columns                      |
07         DW       VRAM page length                 |
09         DW       VRAM start address               |
0B 8h x DW          Cursor row/column for 8 pages    +These are values
1B         DW       Cursor type                      | directly copie d
1D         DB       Active page                      | from low memor y.
1E         DW       CRTC address                     |
20         DB       Port 3D8 data                    |
21         DB       Port 3D9 data                    |
22         DB       Number of rows                   |
23         DW       Point size  ---------------------+
25         DB       Active DCC
26         DB       Auxiliary DCC
27         DW       Number of colors this mode
29         DB       Number of pages this mode
2A         DB       Number of scans this mode (0=200 1 =350 2=400 3=4 80)
2B         DB       Primary character block
2C         DB       Secondary character block
2D         DB       Misc. state info (see table below)
2E 3h x DB          Reserved (set to zero)
31         DB       VRAM size (0=64K 1=128K 2=192K 3=2 56K)
32         DB       Save pointer state info (see table below)
33 Dh x DB          Reserved (set to zero)
```

Table 4-1 Information Block Offset 2D

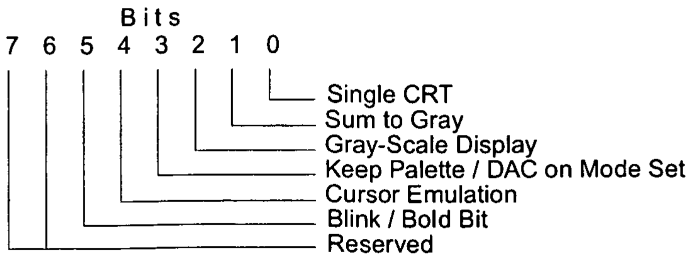
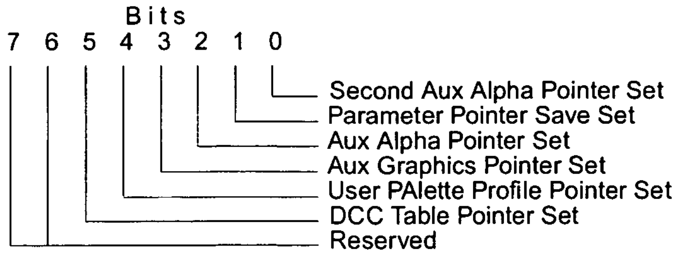


Table 4-2 Information Block Offset 32



Function 1Ch - Save/Restore Video State

Input:	AH=1Ch	Save/Restore Video State
	AL	Function request (0-2)
		00 - Return size of buffer
		01 - Save video state
		02 - Restore video state
	ES:BX	Pointer to buffer
	CX	Bit map of request
		001 - Video h/w status
		010 - Video data arena
		100 - Video DAC and color registers
Return:	AL=1Ch	
	BX	Buffer size if AL input =00

Note
Emulating IBM code this function does not save the latch data or color select register 14h of the Attributes Controller. Because it doesn't get saved or restored, we don't touch it at all.

The format of the save buffer is as follows:

Offsets

00h	DW	Offset to the Video Hardware Status buffer
02h	DW	Offset to the Video Data Arena buffer
04h	DW	Offset to the DAC and Color Registers buffer
06h 0Dh x DW		Reserved (uninitialized)

Video Hardware Status

00h	DB	Sequencer (3C4) index
01h	DB	CRTC (3D4) index
02h	DB	Graphics controller (3CE) index
03h	DB	Attributes controller (3C0) index
04h	DB	Feature control register (3DA) data
05h	4h x DB	Sequencer data
09h	DB	Misc. output register (3C2) data
0Ah	19h x DB	CRTC data
23h	14h x DB	Attributes controller data
37h	9h x DB	Graphics controller data
40h	DW	CRTC base port
42h	4h x DB	Latch data

Video Data Arena

00h	DB	EquipLow AND 30h
01h	DB	CRTMode
02h	DW	CRTColumns
04h	DW	VRAMPageLength
06h	DW	VRAMStartAddr
08h	8h x DW	CursorRowCol
18h	DW	CursorType
1Ah	DB	ActivePage
1Bh	DW	CRTCAddr
1Dh	DB	Port3D8Data
1Eh	DB	Port3D9Data
1Fh	DB	MaxRow
0h	DW	Points
22h	DB	EGAInfo
23h	DB	DIPInfo
24h	DB	VGAInfo
25h	DB	DCCIndex
26h	DD	SavePtr
2Ah	DD	Vec05
2Eh	DD	Vec1D
32h	DD	Vec1F
36h	DD	Vec43

DAC and Color Registers

00h	DB	DAC state (3C7) AND 01h (1=read mode, 0=write mode)
01h	DB	DAC (3C8) index (adjusted)
02h	DB	Pel mask (3C6) data
03h	300h x DB	DAC RGB data

5 VESA VBE FUNCTION SUMMARY

These functions are defined by the Video Electronics Standards Association. They cover issues of inquiry on chip capability, and available modes. The following section shows the SPC8106F0C implementation of these functions. The SPC8106F0C BIOS supports VESA VBE Core functions 1.2, VBE/PM functions 1.0 and VBE/DDC function 1.0. BIOS version 2.0 will support VBE Core functions 2.0.

5.1 Status Information

Every function returns status information in the AX register. The format of the status word is as follows:

AL ==	4Fh:	Function is supported
AL !=	4Fh:	Function is not supported
AH ==	00h:	Function call successful
AH ==	01h:	Function call failed
AH ==	02h:	Software supports this function, but the hardware does not
AH ==	03h:	Function call invalid in current video mode

5.2 VESA Functions

Function 00h - Return VBE Controller Information

Input:	AX	= 4F00h	Return VBE Controller Information
	ES:DI	=	Pointer to buffer in which to place VbeInfoBlock structure (VbeSignature should be set to 'VBE2' when function is called to indicate VBE 2.0 information is desired and the information block is 512 bytes in size.)
Return:	AX	=	VBE Return Status

Note
All other registers are preserved.

Function 01h - Return VBE Mode Information

Input:	AX	= 4F01h	Return VBE mode information
	CX	=	Mode number
	ES:DI	=	Pointer to ModeInfoBlock structure
Return:	AX	=	VBE Return Status

Note
All other registers are preserved.

Function 02h - Set VBE Mode

```
Input:   AX      = 4F02h  Set VBE Mode
         BX      =        Desired Mode to set
                          D0-D8 = Mode number
                          D9-D13 = Reserved (must be 0)
                          D14 = 0 Use windowed frame buffer  model
                              = 1 Use linear/flat frame buffer model
                          D15 = 0 Clear display memory
                              = 1 Don't clear display memory
Return:  AX      =        VBE Return Status
```

Note
All other registers are preserved.

Function 03h - Return current VBE Mode

```
Input:   AX      = 4F03h  Return current VBE Mode
Return:  AX      =        VBE Return Status
         BX      =        Current VBE mode
                          D0-D13 = Mode number
                          D14 = 0 Windowed frame buffer model
                              = 1 Linear/flat frame buffer model
                          D15 = 0 Memory cleared at last mode set
                              = 1 Memory not cleared at last mode set
```

Note
All other registers are preserved.

Function 04h - Save/Restore state

```
Input:   AX      =        4F04h Save/Restore state
         DL      =        00h Return save/restore state buffer size
                 =        01h Save state
                 =        02h Restore state
         CX      =        Requested states
                 D0 =     Save/restore controller hardware state
                 D1=      Save/restore BIOS data state
                 D2=      Save/restore DAC state
                 D3=      Save/restore Register state
         ES:BX   =        Pointer to buffer (if DL <> 00h)
Return:  AX      =        VBE Return Status
         BX      =        Number of 64-byte blocks to hold the state buffer (if
                          DL=00h)
```

Note
All other registers are preserved.

Function 05h - Display Window Control

Input:	AX	= 4F05h	VBE Display Window Control
	BH	= 00h	Set memory window
		= 01h	Get memory window
	BL	=	Window number
		= 00h	Window A
		= 01h	Window B
	DX	=	Window number in video memory in window granularity units (Set Memory Window only)
Return:	AX	=	VBE Return Status
	DX	=	Window number in window granularity units (Get Memory Window only)

Function 06h - Set/Get Logical Scan Line Length

Input:	AX	= 4F06h	VBE Set/Get Logical Scan Line Length
	BL	= 00h	Set Scan Line Length in Pixels
		= 01h	Get Scan Line Length
		= 02h	Set Scan Line Length in Bytes
		= 03h	Get Maximum Scan Line Length
	CX	=	If BL=00h Desired Width in Pixels If BL=02h Desired Width in Bytes (Ignored for Get Functions)
Return:	AX	=	VBE Return Status
	BX	=	Bytes Per Scan Line
	CX	=	Actual Pixels Per Scan Line (truncated to nearest complete pixel)
	DX	=	Maximum Number of Scan Lines

Function 07h - Set/Get Display Start

Input:	AX	= 4F07h	VBE Set/Get Display Start Control
	BH	= 00h	Reserved and must be 00h
	BL	= 00h	Set Display Start
		= 01h	Get Display Start
		= 80h	Set Display Start during Vertical Retrace
	CX	=	First Displayed Pixel In Scan Line (Set Display Start only)
	DX	=	First Displayed Scan Line (Set Display Start only)
Return:	AX	=	VBE Return Status
	BH	=	00h Reserved and will be 0 (Get Display Start only)
	CX	=	First Displayed Pixel In Scan Line (Get Display Start only
	DX	=	First Displayed Scan Line (Get Display Start only)

Function 08h - Set/Get DAC Palette Format

Input:	AX	=	4F08h	VBE Set/Get Palette Format
	BL	=	00h	Set DAC Palette Format
		=	01h	Get DAC Palette Format
	BH	=		Desired bits of color per primary (Set DAC Palette Format only)
Return:	AX	=		VBE Return Status
	BH	=		Current number of bits of color per primary

Function 09h - Set/Get Palette Data

Input:	AX	=	4F09h	VBE Load/Unload Palette Data
	BL	=	00h	Set Palette Data
		=	01h	Get Palette Data
		=	02h	Set Secondary Palette Data
		=	03h	Get Secondary Palette Data
		=	80h	Set Palette Data during Vertical Retrace with Blank Bit on
	CX	=		Number of palette registers to update
	DX	=		First palette register to update
	ES:DI	=		Table of palette values (see below for format)
Return:	AX	=		VBE Return Status

Format of Palette Values: Alignment byte, Red byte, Green byte, Blue byte

Function 0Ah - Return VBE Protected Mode Interface

Input:	AX	=	4F0Bh	VBE 2.0 Protected Mode Interface
	BL	=	00h	Return protected mode table
Return:	AX	=		Status
	ES	=		Real Mode Segment of Table
	DI	=		Offset of Table
	CX	=		Length of Table including protected mode code (for copying purposes)

The format of the table is as follows:

ES:DI + 00h	Word Offset in table of Protected mode code for Function 5 for Set Window Call
ES:DI + 02h	Word Offset in table of Protected mode code for Function 7 for set Display Start
ES:DI + 04h	Word Offset in table of Protected mode code for Function 9 for set Primary Palette data
ES:DI + 06h	Word Offset in table of Ports and Memory Locations that the application may need I/O privilege for. (Optional: if unsupported this must be 0000h) (See Sub-table for format)
ES:DI + ?	Variable remainder of Table including Code

Function 10h - Display Power Management Extensions
(valid for BIOS version 1.2 or greater)

The VESA VBE sub-Function 10h is used to implement the VBE / PM services. The VBE / PM services are defined as follows:

Sub-Function 00h - Report VBE / PM Capabilities

```
Input:   AH      = 4Fh   VESA Extension.
         AL      = 10h   VBE/PM Services.
         BL      = 00h   Report VBE/PM Capabilities.
         CX      = 00h   Controller unit number (00 = primary controller).
         ES:DI           Null pointer, must be 0000:0000h in version 1.0.
                         Reserved for future use.
Return:  AX      =       Status.
         BH      =       Power saving state signals supported by the
                         controller. 1 = supported, 2 = not supported
                         bit 0 STANDBY
                         bit 1 SUSPEND
                         bit 2 OFF
                         bit 3 REDUCED ON
                         bits 4-7 reserved for future power control of the
                                  display controller or other related
                                  circuits.
         BL      =       VBE/PM version number (0001 0000b for this version).
                         bits 0-3 minor version number
                         bits 4-7 major version number
         CX      =       Unchanged
         ES:DI           Unchanged
All other registers may be destroyed.
```

Sub-Function 01h - Set Display Power State

```
Input:   AH      = 4Fh   VESA Extension.
         AL      = 10h   VBE/PM Services.
         BL      = 01h   Set Display Power State.
         BH      = 00h   ON
                 = 01h   STANDBY
                 = 02h   SUSPEND
                 = 04h   OFF
                 = 08h   REDUCED ON
                 All other BH values are currently undefined and are reserved
                 for future power control of the display controller.
         CX      = 00h   Controller unit number (00 = primary controller).
Return:  AX      =       Status.
         BH      =       Unchanged
         CX      =       Unchanged
All other registers may be destroyed.
```

Sub-Function 02h - Get Display Power State

```
Input:   AH       = 4Fh   VESA Extension.
         AL       = 10h   VBE/PM Services.
         BL       = 02h   Get Display Power State.
         CX       = 00h   Controller unit number (00 = primary controller).
Return:  AX       =       Status.
         BH               Power state currently requested by the controller .
                  = 00h   ON
                  = 01h   STANDBY
                  = 02h   SUSPEND
                  = 04h   OFF
                  = 08h   REDUCED ON
                  All other BH values are reserved and may be used  to signal
                  other power saving states in future revisions of  VBE/PM. For
                  future compatibility, applications written for VBE/PM 1.0
                  should ignore the value of bits 4 to 7.
         CX       =       Unchanged
All other registers may be destroyed.
```

Function 15h - Display Identification Extensions
(valid for BIOS version 1.2 or greater)

The VESA VBE sub-Function 15h is used to implement the VBE / DDC services. The VBE / DDC services are defined as follows:

Sub-Function 00h - Report VBE / DDC Capabilities

```
Input:   AH       = 4Fh   VESA Extension.
         AL       = 15h   VBE/DDC Services.
         BL       = 00h   Report VBE/DDC Capabilities.
         CX       = 00h   Controller unit number (00 = primary controller).
         ES:DI            Null pointer, must be 0000:0000h in version 1.0.
                          Reserved for future use.
Return:  AX       =       Status
         BH       =       Approximate time, in seconds, rounded up, to transfer
                          one EDID block (128 bytes).
         BL       =       DDC level supported.
                          bit 0 = 0 DDC1 not supported
                                = 1 DDC1 supported
                          bit 1 = 0 DDC2 not supported
                                = 1 DDC2 supported
                          bit 2 = 0 screen not blanked during data transfer
                                = 1 screen blanked during data transfer
         CX       =       Unchanged
         ES:DI            Unchanged
All other registers may be destroyed.
```

Sub-Function 01h - Read EDID

Input:	AH	=	4Fh	VESA Extension.
	AL	=	15h	VBE/DDC Services.
	BL	=	01h	Read EDID.
	CX	=	00h	Controller unit number (00 = primary controller).
	DX	=	00h	EDID block number. Zero is the only valid value in version 1.0.
	ES:DI			Pointer to the area in which the EDID block (128 bytes) shall be returned.
Return:	AX	=		Status
	BH	=		Unchanged
	CX	=		Unchanged
	ES:DI			Pointer to the area in which the EDID block (128 bytes) is returned.

All other registers may be destroyed.

Sub-Function 02h - Read VDIF Block

Input:	AH	=	4Fh	VESA Extension.
	AL	=	15h	VBE/DDC Services.
	BL	=	02h	Read VDIF block.
	CX	=	00h	Controller unit number (00 = primary controller).
	DX	=	00h	VDIF block number (64 byte block).
	ES:DI			Pointer to the area in which the VDIF block (64 bytes) shall be returned.
Return:	AX	=		Status
	BH	=		Unchanged
	CX	=		Unchanged
	ES:DI			Pointer to the area in which the VDIF block (64 bytes) is returned.

All other registers may be destroyed.

6 SOLLEX FUNCTION SUMMARY

These functions are defined by Seiko Epson as a generic interface for functions not covered by the Video Electronics Standards Association or IBM's standard video BIOS. The following has been marked up to show the SPC8106F0C implementation of these functions.

6.1 Sollex Status Information

Every function returns status information in the AX register. The format of the status words is as follows:

```
AL == 7Fh:    Function is supported
AL != 7Fh:    Function is not supported
AH == 00h:    Function call successful
AH == 01h:    Function call fails
```

6.2 Sollex Reserved Bits

All reserved bit returns will return 0 by default, unless otherwise noted.

6.3 Sollex Functions

Function 00h - Return Extensions Info

Not supported in SPC8106F0C.

Function 01h - Adapter Control

```
Input:     AH=7Fh    SOLLEX Support
           AL=01h    Adapter Control
           BL=00h    Set Adapter
           CX        Adapter Request
Return:    AX        Status

Input:     AH=7Fh    SOLLEX Support
           AL=01h    Adapter Control
           BL=01h    Get Adapter
Return:    AX        Status
           BX        Adapter type
           DX        Display type

Input:     AH=7Fh    SOLLEX Support
           AL=01h    Adapter Control
           BL=02h    Return Adapter Support
           CX        Adapter Request
Return:    AX        Status
```

Table 6-1 Sub-Function 00h: Set Adapter

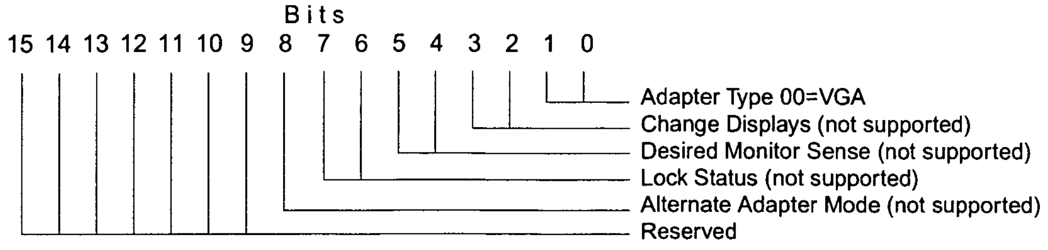
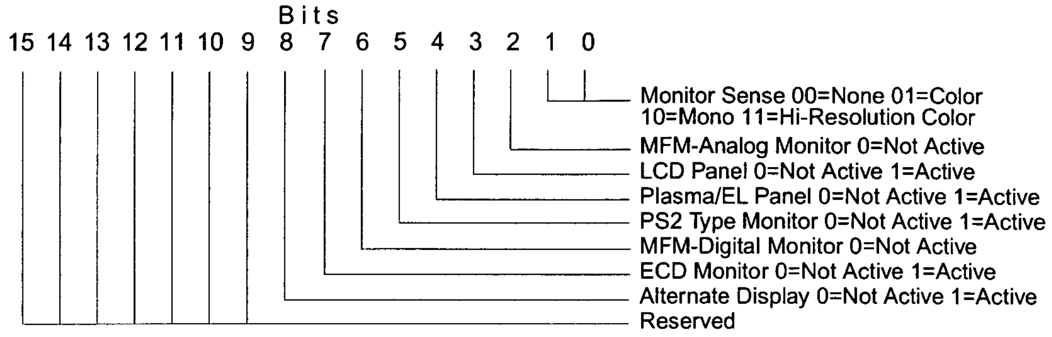

Bits
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Adapter Type 00=VGA
Change Displays (not supported)
Desired Monitor Sense (not supported)
Lock Status (not supported)
Alternate Adapter Mode (not supported)
Reserved

Table 6-2 Sub-Function 01h: Get Adapter


Bits
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Monitor Sense 00=None 01=Color
10=Mono 11=Hi-Resolution Color
MFM-Analog Monitor 0=Not Active
LCD Panel 0=Not Active 1=Active
Plasma/EL Panel 0=Not Active 1=Active
PS2 Type Monitor 0=Not Active 1=Active
MFM-Digital Monitor 0=Not Active
ECD Monitor 0=Not Active 1=Active
Alternate Display 0=Not Active 1=Active
Reserved

SubFunction 2 uses the same Adapter Request format as SubFunction 0 to determine whether the requested adapter setting could be successfully set in the current environment.

Function 02h - Display Output Control

```
Input:     AH=7Fh    SOLLEX Support
           AL=02h    Display Output Control
           BL=00h    Set Display Output
           CX        Display Setting
Return:    AX        Status

Input:     AH=7Fh    SOLLEX Support
           AL=02h    Display Output Control
           BL=01h    Get Display Output
Return:    AX        Status
           BX        Display Output
           CX        Displays attached
```

Table 6-3 Sub-Function 00h: Set Display Output

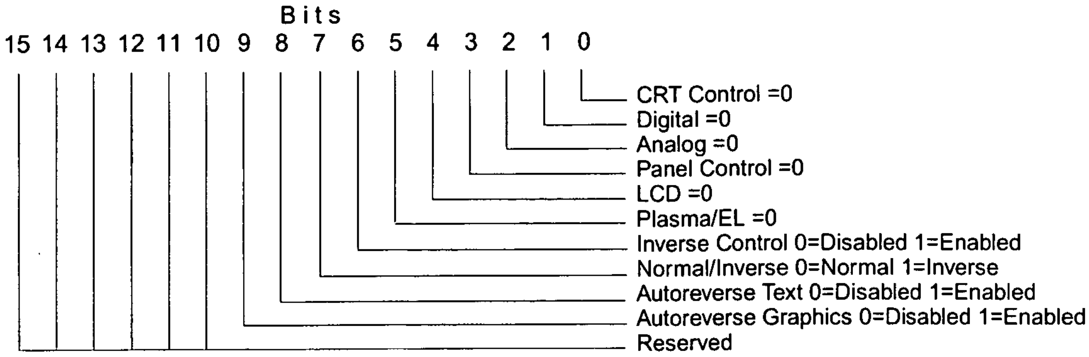
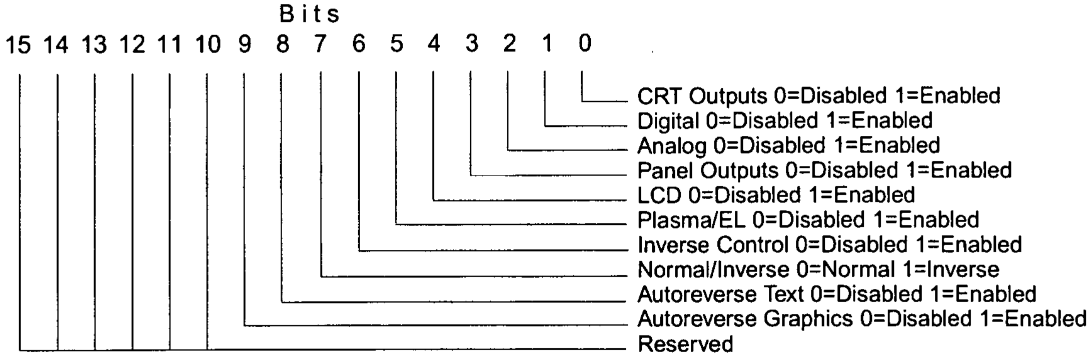
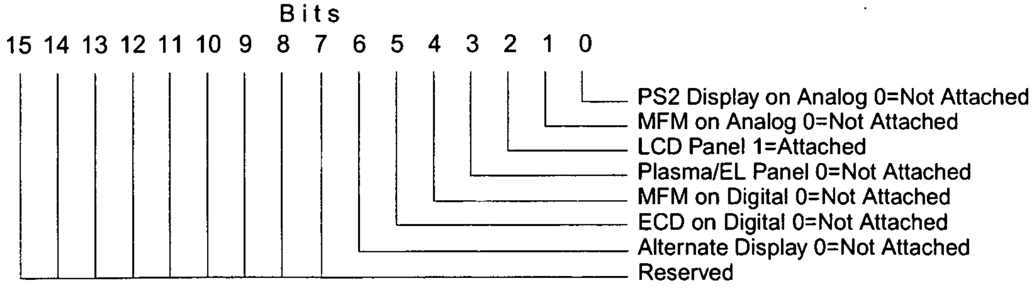

Bits
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
CRT Control =0
Digital =0
Analog =0
Panel Control =0
LCD =0
Plasma/EL =0
Inverse Control 0=Disabled 1=Enabled
Normal/Inverse 0=Normal 1=Inverse
Autoreverse Text 0=Disabled 1=Enabled
Autoreverse Graphics 0=Disabled 1=Enabled
Reserved

Table 6-4 Sub-Function 01h: Get Display Output


Bits
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
CRT Outputs 0=Disabled 1=Enabled
Digital 0=Disabled 1=Enabled
Analog 0=Disabled 1=Enabled
Panel Outputs 0=Disabled 1=Enabled
LCD 0=Disabled 1=Enabled
Plasma/EL 0=Disabled 1=Enabled
Inverse Control 0=Disabled 1=Enabled
Normal/Inverse 0=Normal 1=Inverse
Autoreverse Text 0=Disabled 1=Enabled
Autoreverse Graphics 0=Disabled 1=Enabled
Reserved

The CX register contains the following bit mask that describes the attached displays:

Table 6-5 Display Attached


Bits
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
PS2 Display on Analog 0=Not Attached
MFM on Analog 0=Not Attached
LCD Panel 1=Attached
Plasma/EL Panel 0=Not Attached
MFM on Digital 0=Not Attached
ECD on Digital 0=Not Attached
Alternate Display 0=Not Attached
Reserved

Function 03h - Video Support Control

Not supported in SPC8106F0C.

Function 04h - Power Control

```
Input:    AH=7Fh    SOLLEX Support
          AL=04h    Power Control
          BL=00h    Set Power State
          CX        Power State (0 to Maximum State)
Return:   AX        Status

Input:    AH=7Fh    SOLLEX Support
          AL=04h    Power Control
          BL=01h    Get Power State
Return:   AX        Status
          CX        Power State
          DX        Maximum State

Input:    AH=7Fh    SOLLEX Support (UNSUPPORTED in SPC8106F0 C)
          AL=04h    Power Control
          BL=02h    Set Time Out Reset
          CX        Time Out Reset (0 to Maximum Time Out  Reset)
Return:   AX        Status AL=7Fh, AH=01h

Input:    AH=7Fh    SOLLEX Support (UNSUPPORTED in SPC8106F0 C)
          AL=04h    Power Control
          BL=03h    Get Time Out Reset
Return:   AX        Status AL=7Fh, AH=01h
```

SubFunction 00h: Set Power State

Set Power State according to table below.

SubFunction 01h: Get Power State

Returns Power State according to table below.

	State 0	State 1	State 2	State 3	State 4	State 5***
Clock	Yes	Yes	Yes	No	Yes*	Yes****
Display (panel) enabled	Yes	No	No	No	No	Yes
CRT display access	Yes	No	No	No	No	Yes
CPU to VRAM Refresh	Yes	Yes	No	No	No	Yes
VRAM Refresh	Yes	Yes	Yes	No	External	Yes
IO Write/IO Read	Yes	Yes	Yes	Aux Registers	Aux Registers	Yes
Ext.RamDAC	Yes	No**	No**	No	No	No
Relative Power Saving	None	Low	Medium	High	High	minimal

* can use Power Down clock if available
** read/write allowed
*** not available on CRT
**** Clock is slower by 25%

Function 05h - Load Register

Not supported in SPC8106F0C.

Function 06h - Multiple Font Control

```
Input:    AH=7Fh    SOLLEX Support
          AL=06h    Multiple Font Control
          BL=00h    Set Multiple Font State
          CX        Multiple Font State
                    0h=off
                    1h=on
Return:   AX        Status

Input:    AH=7Fh    SOLLEX Support
          AL=06h    Multiple Font Control
          BL=01h    Get Multiple Font State
Return:   AX        Status
          BL        Multiple Font State
                    0h=off
                    1h=on
```

Function 07h - Fill Video RAM

Not supported in SPC8106F0C.

Function 08h - Autocenter Control

```
Input:    AH=7Fh    SOLLEX Support
          AL=08h    Autocenter control
          BL=00h    Set Autocenter control
          CX        0000h Disable Autocenter
                    0001h Enable Autocenter
Return:   AX        Status

Input:    AH=7Fh    SOLLEX Support
          AL=08h    Autocenter control
          BL=01h    Get Autocenter state
Return:   AX        Status
          BL        Autocenter control status
                    00h=Disabled
                    01h=Enabled
```

Function 09h - Lookup Table Control

Not supported in SPC8106F0C.

Function 0Ah - Non-Standard Font Control (SPC8106F0C uses 19pt.font for h/w text expansion*)

Input:	AH=7Fh	SOLLEX Support
	AL=0Ah	Non-Standard Font Control
	BL=00h	Set Non-Standard Font Control state
	CL	Font Width
	CH	Font Height (if CX=0, it will set normal system font to be default)
Return:	AX	Status

Input:	AH=7Fh	SOLLEX Support
	AL=0Ah	Non-Standard Font Control
	BL=01h	Get Non-Standard Font Control state
Return:	AX=	Status
	BL=	Font Width
	BH=	Font Height (normal system font will return proper values not 0 as in the set)
	ES:DI	pointer to table of available fonts (format width, height, width, height ... 00, 00)

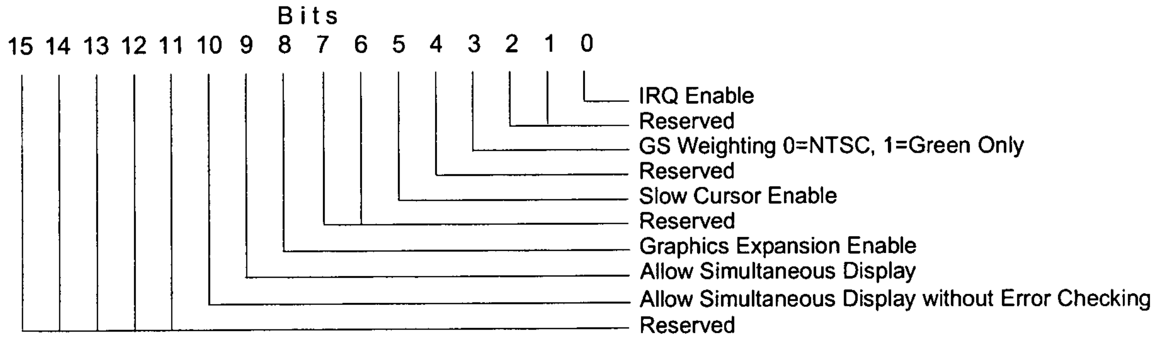
* use 19 point for set to enable Hardware Text Expansion

Function FFh - Chip Specific Function

```
Input:    AH=7Fh   SOLLEX Support
          AL=FFh   Chip Specific Function
          BL=00h   Set
          CX       Requested State
          DH       Graphic expansion line number (if D8  of CX is 1)
Return:   AX       Status

Input:    AH=7Fh   SOLLEX Support
          AL=FFh   Chip Specific Function
          BL=01h   Get
Return:   AX=      Status
          BH=      Chip Revision Product Code
          BL=      Chip Revision
          CX=      Current State
          DH=      Graphic expansion line number (if D8  of CX is 1)
```

Table 6-6 Chip Specific Function


Bits
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
IRQ Enable
Reserved
GS Weighting 0=NTSC, 1=Green Only
Reserved
Slow Cursor Enable
Reserved
Graphics Expansion Enable
Allow Simultaneous Display
Allow Simultaneous Display without Error Checking
Reserved

7 PHYSICAL LAYOUT OF THE BIOS

7.1 Single Segment Structure

The BIOS will exist in a 32 KB section of ROM located at C000 or any other configurable location. Segment references within the code will self define at initialization time, however an initial BIOS segment must be configured into the ROM image.

The layout of this segment is as follows:

Initialization Routine	Function Dispatch	Function Calls	Video Parameter Table	Character Fonts 8, 14, 16, & 18 or 19 if needed	Extensions Initialization	Extensions Dispatch	Extensions Function Calls	Extensions Parameter Tables

The single segment structure of the BIOS versus a dual image structure (where the Extensions have their own segment) has the advantage that OS/2 2.0 will work correctly in the virtual DOS box.

7.2 IBM Notice, Copyright Notice

Some software expects the word 'IBM' to be located at C000:001E to determine if it is running on an EGA or better. We have initially put the word 'IBM' at this location, but, if the BIOS is relocated to another segment, E000 for example, software that does check at C000:001E will not work. There is very little chance of running across current software with this test.

The BIOS will also contain the string “Copyright (c) Seiko Epson Corp. 1987, 1993. All rights reserved” in two locations. One of these must stay in the code, the other is part of the header that is displayed on power-up. The power-up string is configurable and can be overwritten by the OEM.